UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/975,587	10/11/2001	Dean Bernard Jacobs	BEAS-01077US3	8688

23910 7590 07/25/2007

FLIESLER MEYER LLP
650 CALIFORNIA STREET
14TH FLOOR
SAN FRANCISCO, CA 94108

EXAMINER	
BATURAY, ALICIA	
ART UNIT	**PAPER NUMBER**
2155	

MAIL DATE	DELIVERY MODE
07/25/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	09/975,587	JACOBS ET AL.	
	Examiner	Art Unit	
	Alicia Baturay	2155	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *18 May 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-51* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-51* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *11 October 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *05/18/2007*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This Office Action is in response to the amendment filed 18 May 2007.

2. Claims 1, 36, 40, 41 and 46-51 were amended.

3. Claims 1-51 are pending in this Office Action.

Response to Amendment

4. Applicant's amendments and arguments with respect to claims 1-51 filed on 18 May 2007 have been fully considered but they are deemed to be moot in view of the new grounds of rejection.

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 1-5, 7-20 and 24-51 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gehani et al. (U.S. 5,765,171) and further in view of Britton et al. (U.S. 5,613,060).

Gehani teaches the invention substantially as claimed including a replication protocol which includes associating a database version vector with each copy of the database in the system, the database version vectors of the replicas are compared to efficiently determine if update replication is necessary (see Abstract).

7. With respect to claim 1, Gehani teaches a computer-based system for replicating data over a network, comprising: a master server containing an original copy of the data, the master server comprising: a master user layer to start a data replication process by calling a start method, the master user layer further adapted to send information relating to the original copy of the data; a master service layer, containing the start method to receive the call from the master user layer and the information relating to the original copy of the data, the master service layer and to create and send a data replication packet containing at least some of the information relating to the original copy of the data (Gehani, col. 7, lines 44-46); a slave server to store a copy of the data from the master server, the slave server comprising: a slave service layer to receive the data replication packet from the master service layer and process the data replication packet, and to send information relating to the data replication packet; and a slave user layer to receive the information relating to the data replication packet from the slave service layer, and to store the information in the data replication packet (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously

(Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46), wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

8. With respect to claim 2, Gehani teaches the invention the invention described in claim 1, including a system where the master user layer is in communication with at least one of a master user and a master user device (Gehani, col. 4, lines 28-33).

9. With respect to claim 3, Gehani teaches the invention the invention described in claim 1, including a system where the master user layer sends information relating to the original copy of the data in the form of a delta, the delta containing information relating to changes between a previous state and the current state of the original copy of the data (Gehani, col. 7, lines 44-46).

10. With respect to claim 4, Gehani teaches the invention the invention described in claim 1, including a system where the master user layer updates the original copy of the data (Gehani, col. 7, lines 15-17).

11. With respect to claim 5, Gehani teaches the invention the invention described in claim 1, including a system where the master user layer sends a roll-back message indicating that a change to the original copy of the data should not be replicated on a slave server (Gehani, col. 7, lines 44-57).

12. With respect to claim 7, Gehani teaches the invention the invention described in claim 1, including a system where the master user layer creates a delta between the present state of the original copy of the data and the prior state of the original copy of the data (Gehani, Fig. 3; col. 5, lines 37-46).

13. With respect to claim 8, Gehani teaches the invention the invention described in claim 1, including a system where the master user layer creates a delta between the present state of the original copy of the data and a previous state of the original copy of the data (Gehani, Fig. 3; col. 5, lines 37-46).

14. With respect to claim 9, Gehani teaches the invention the invention described in claim 1, including a system where the master user layer generates a unique version number for each state of the original copy of the data (Gehani, col. 5, lines 60-62).

15. With respect to claim 10, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer multicasts the data replication packet (Gehani, col. 7, lines 3-5).

16. With respect to claim 11, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer heartbeats the data replication packet (Gehani, col. 7, lines 3-17).

17. With respect to claim 12, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer includes a version number in the data replication packet (Gehani, col. 7, lines 44-46).

18. With respect to claim 13, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer includes information necessary to update the copy of the data on the slave server to the current state of the original copy of the data (Gehani, col. 7, lines 44-60).

19. With respect to claim 14, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer creates and send a data replication packet comprising a delta (Gehani, col. 7, lines 44-60).

20. With respect to claim 15, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer creates and sends a data replication packet comprising a delta between successive states of the original copy of the data (Gehani, Fig. 6; col. 8, lines 32-58).

21. With respect to claim 16, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer creates and sends a data replication packet comprising a delta between arbitrary states of the original copy of the data (Gehani, col. 7, lines 44-46).

22. With respect to claim 17, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer requests a delta from the master user layer (Gehani, col. 7, lines 44-46).

23. With respect to claim 18, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer sends a commit message to a slave service layer (Gehani, col. 7, lines 44-46).

24. With respect to claim 19, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer heartbeats a commit message to a slave service layer (Gehani, col. 7, lines 3-17).

25. With respect to claim 20, Gehani teaches the invention the invention described in claim 1, including a system where the master service layer multicasts a commit message to a slave service layer (Gehani, col. 7, lines 3-5).

26. With respect to claim 24, Gehani teaches the invention the invention described in claim 1, including a system where the slave user layer is in communication with at least one of a slave user and a slave user device (Gehani, col. 4, lines 14-27).

27. With respect to claim 25, Gehani teaches the invention the invention described in claim 1, including a system where the slave user layer checks the current version number of data stored on the slave server (Gehani, col. 7, lines 46-60).

28. With respect to claim 26, Gehani teaches a system where the slave user layer commits information relating to the data replication packet to the data stored on the slave server (Gehani, col. 7, lines 46-60).

29. With respect to claim 27, Gehani teaches the invention the invention described in claim 1, including a system where the slave user layer aborts an update to the data stored on the slave server (Gehani, col. 7, lines 46-60).

30. With respect to claim 28, Gehani teaches the invention the invention described in claim 1, including a system where the slave user layer processes a prepare request contained in the data replication packet (Gehani, col. 7, lines 46-60).

31. With respect to claim 29, Gehani teaches the invention the invention described in claim 1, including a system where the slave user layer sends a response to the slave service layer relating to a prepare request contained in the data replication packet (Gehani, col. 7, lines 46-60).

32. With respect to claim 30, Gehani teaches the invention the invention described in claim 1, including a system where the slave user layer persistently caches data on a local disk (Gehani, col. 7, lines 46-60).

33. With respect to claim 31, Gehani teaches the invention the invention described in claim 1, including a system where the slave user layer updates the version number of the copy of the data on the slave server (Gehani, col. 7, lines 46-60).

34. With respect to claim 32, Gehani teaches the invention the invention described in claim 1, including a system where the slave service layer requests a delta from the master service layer (Gehani, col. 4, lines 41-45).

35. With respect to claim 33, Gehani teaches the invention the invention described in claim 1; including a system where the slave service layer requests the current version number of the data stored on the slave server from the slave user layer (Gehani, col. 7, lines 46-60).

36. With respect to claim 34, Gehani teaches the invention the invention described in claim 1, including a system where the slave service layer sends a commit message to the slave user layer (Gehani, col. 7, lines 44-46).

37. With respect to claim 35, Gehani teaches the invention the invention described in claim 1, including a system where the slave service layer sends an abort message to the slave user layer (Gehani, col. 7, lines 54-57).

38. With respect to claim 36, Gehani teaches a computer-based method for replicating data from a master server to a slave server, comprising: sending a start call from a master user level to a master service level on a master server, the start call containing information relating to the current state of master data on the master server; sending the information to a slave service layer on a slave server (Gehani, col. 7, lines 44-46), the slave service layer adapted to check a slave user layer on the slave server to determine whether slave data on the slave server has the current state (Gehani, col. 7, lines 46-48); sending a request for a delta from the slave service layer to the master service layer, the master service layer adapted to request and receive a delta from the master user layer (Gehani, col. 4, lines 41-45); sending a delta from the master service layer to the slave service layer, the delta containing the

information necessary to bring the slave data up to the current state, the slave service layer adapted to process the delta and send the information to the slave user layer; and updating the slave data using the slave user layer (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

39. With respect to claim 37, Gehani teaches the invention the invention described in claim 36, including a method further comprising: determining a version number for the current state of the data using the master user layer (Gehani, col. 7, lines 44-46).

40. With respect to claim 38, Gehani teaches the invention the invention described in claim 36, including a method further comprising: sending the information to the slave service layer by multicasting (Gehani, col. 7, lines 3-5).

41. With respect to claim 39, Gehani teaches the invention the invention described in claim 36, including a method further comprising: sending information to the slave service layer, the information comprising a version number for the current state of the master data (Gehani, col. 7, lines 44-46).

42. With respect to claim 40, Gehani teaches a computer-based method for replicating data from a master server to a slave server, comprising: sending a new delta from a master user level to a master service level on a master server, the delta containing information relating to a change from the prior state to the current state in master data stored on the master server; sending the new delta form the master service layer to a slave service layer on a slave server (Gehani, col. 7, lines 44-46), the slave service layer adapted to check a slave user layer on the slave server to determine whether the slave data on the slave server has the current state (Gehani, col. 7, lines 46-48); sending a request for a syncing delta from the slave service layer to the master service layer (Gehani, col. 4, lines 41-45), the master service layer adapted to request and receive a syncing delta from the master user layer, the syncing delta containing information necessary to update the slave data to the prior state of the master data; sending the syncing delta from the master service layer to the slave service layer, the slave service layer adapted to process the delta and send the information to the slave user layer to

be committed to the slave data; and committing the information in the new delta to the slave data using the slave user layer (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

43. With respect to claim 41, Gehani teaches a computer-based method for replicating data from a master server to a slave server over a network, the method comprising the steps of: sending a version number from a master service layer to a slave service layer relating to the present state of the original copy of the data on the master server (Gehani, col. 7, lines 44-

46); allowing a slave user layer to determine whether the data on the slave server has been updated to correspond to the version number (Gehani, col. 7, lines 46-48); and requesting a delta be sent from the master service layer to the slave service layer if the data on the slave server does not correspond to the version number (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

44. With respect to claim 42, Gehani teaches a method according to claim 36, further comprising: allowing the slave user layer to persistently cache the data on a local disk for each slave server (Gehani, col. 7, lines 46-60).

45. With respect to claim 43, Gehani teaches a method according to claim 36, further comprising: allowing the master user layer to determine a unique version number for the current state of the data on the master server (Gehani, col. 5, lines 60-62).

46. With respect to claim 44, Gehani teaches a method according to claim 36, further comprising: including data with the version number that is necessary for a slave user layer to update the data on a slave server (Gehani, col. 7, lines 44-46).

47. With respect to claim 45, Gehani teaches a method according to claim 36, further comprising: committing the data necessary to update the slave server as soon as it is received by the slave user layer (Gehani, col. 7, lines 44-46).

48. With respect to claim 46, Gehani teaches a computer-based method for replicating data over a network including a master server and at least one slave server, the method comprising the steps of: sending a packet of information from a master service layer to a slave service layer on each slave server on the network, the information relating to a change in the data stored on the master server and containing a prior version number for the prior state and a new version number for the new state of the data, the information further relating to previous

changes in the data and a previous version number for each previous change (Gehani, col. 7, lines 44-46); allowing a slave user layer on each slave server to determine whether the data on the slave server corresponds to the prior version number contained in the packet (Gehani, col. 7, lines 46-48); allowing each slave user layer to commit the packet of information if the data on the slave server corresponds to the prior version number contained in the packet, the commit also updating the version of the slave server to the new version number; and allowing each slave user layer not corresponding to the prior version number to request that a delta be sent from the master service layer to the slave service layer corresponding to that slave user layer, the delta containing the information necessary to update the slave to the prior version number before the slave service layer commits the packet of information (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-

phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

49. With respect to claim 47, Gehani teaches a computer-based method for replicating data from a master server to at least one slave server over a network, the method comprising the steps of: sending a packet of information from a master service layer on the master server to the user service layer on a slave server, the information relating to a change in the data stored on the master server and containing a version number for the present state of the data (Gehani, col. 7, lines 44-46); allowing the slave user layer on the server to determine whether the slave server has been updated to correspond to the version number contained in the packet, and to further determine whether the slave user layer can process the packet of information if needed to update to correspond to the version number contained in the packet (Gehani, col. 7, lines 46-48); sending a signal from the slave service layer to the master service layer, the signal indicating whether the slave server needs to be updated and whether the slave server can process the update (Gehani, col. 4, lines 41-45); sending a response signal from the master service layer to the slave service layer indicating whether the slave user layer should commit to the information contained in the packet; and committing the packet of information to the slave server if so indicated by the response signal (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

50. With respect to claim 48, Gehani teaches a computer-readable medium, comprising: means for sending a version number from a master service layer to a slave service layer relating to the present state of the original copy of the data on the master server (Gehani, col. 7, lines 44-46); means for allowing a slave user layer to determine whether the data on the slave server has been updated to correspond to the version number (Gehani, col. 7, lines 46-48); and means for requesting a delta be sent from the master service layer to the slave

service layer if the data on the slave server does not correspond to the version number (Gehani, col. 4, lines 41-45).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

51. With respect to claim 49, Gehani teaches a computer program product for execution by a server computer for replicating data from a master server to a slave server over a network, comprising: computer code for sending a version number from a master service layer to a slave service layer relating to the present state of the original copy of the data on the master

server (Gehani, col. 7, lines 44-46); computer code for allowing a slave user layer to determine whether the data on the slave server has been updated to correspond to the version number (Gehani, col. 7, lines 46-48); and computer code for requesting a delta be sent from the master service layer to the slave service layer if the data on the slave server does not correspond to the version number (Gehani, col. 4, lines 41-45).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) and wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

52. With respect to claim 50, Gehani teaches a s computer-based system for replicating data over a network, comprising: means for sending a version number from a master service layer to a slave service layer relating to the present state of the original copy of the data on the master server (Gehani, col. 7, lines 44-46); means for allowing a slave user layer to determine whether the data on the slave server has been updated to correspond to the version number (Gehani, col. 7, lines 46-48); and means for requesting a delta be sent from the master service layer to the slave service layer if the data on the slave server does not correspond to the version number (Gehani, col. 4, lines 41-45).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

53. With respect to claim 51, Gehani teaches a computer system comprising: a processor; object code executed by the processor, the object code configured to: send a version number from a master service layer to a slave service layer relating to the present state of the original copy of the data on the master server (Gehani, col. 7, lines 44-46); allow a slave user layer to determine whether the data on the slave server has been updated to correspond to the version number (Gehani, col. 7, lines 46-48); and request a delta be sent from the master service layer to the slave service layer if the data on the slave server does not correspond to the version number (Gehani, col. 4, lines 41-45).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46) wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while

avoiding extensive delays in the operation of an application program that initiated the commit procedure.

54. Claim 6 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gehani in view of Britton and further in view of Mosher et al. (U.S. 6,785,696).

55. With respect to claim 6, Gehani teaches the invention described in claim 1, including a computer-based system for replicating data over a network, comprising: a master server containing an original copy of the data, the master server comprising: a master user layer to start a data replication process by calling a start method, the master user layer further adapted to send information relating to the original copy of the data; a master service layer, containing the start method to receive the call from the master user layer and the information relating to the original copy of the data, the master service layer and to create and send a data replication packet containing at least some of the information relating to the original copy of the data (Gehani, col. 7, lines 44-46); a slave server to store a copy of the data from the master server, the slave server comprising: a slave service layer to receive the data replication packet from the master service layer and process the data replication packet, and to send information relating to the data replication packet; and a slave user layer to receive the information relating to the data replication packet from the slave service layer, and to store the information in the data replication packet (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46), wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

The combination of Gehani and Britton does not explicitly teach the use of a timeout value for replication.

However, Mosher teaches a system where the master user layer is adapted to set a timeout value for the replication (Mosher, Fig. 5B; col. 7, lines 4-10).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combination of Gehani and Britton in view of Mosher in order to enable the use of a timeout value for replication. One would be motivated to do so in order to allow for a proper synchronization of the database files.

56. Claims 21-23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gehani in view of Britton and further in view of Fiske (U.S. 6,324,692).

57. With respect to claim 21, Gehani teaches the invention described in claim 1, including a computer-based system for replicating data over a network, comprising: a master server containing an original copy of the data, the master server comprising: a master user layer to start a data replication process by calling a start method, the master user layer further adapted to send information relating to the original copy of the data; a master service layer, containing the start method to receive the call from the master user layer and the information relating to the original copy of the data, the master service layer and to create and send a data replication packet containing at least some of the information relating to the original copy of the data (Gehani, col. 7, lines 44-46); a slave server to store a copy of the data from the master server, the slave server comprising: a slave service layer to receive the data replication packet from the master service layer and process the data replication packet, and to send information relating to the data replication packet; and a slave user layer to receive the information relating to the data replication packet from the slave service layer, and to store the information in the data replication packet (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master

service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46), wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

The combination of Gehani and Britton does not explicitly teach the master server sending the slave server an abort message.

However, Fiske teaches a system where the master service layer is adapted to send an abort message to a slave service layer (Fiske, col. 5, lines 37-42).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combination of Gehani and Britton in view of Fiske in order to enable the master server sending the slave server an abort message. One would be motivated to do so in order to allow for the master server to correct potential mistakes encountered during replication.

58. With respect to claim 22, Gehani teaches the invention described in claim 1, a computer-based system for replicating data over a network, comprising: a master server containing an

original copy of the data, the master server comprising: a master user layer to start a data replication process by calling a start method, the master user layer further adapted to send information relating to the original copy of the data; a master service layer, containing the start method to receive the call from the master user layer and the information relating to the original copy of the data, the master service layer and to create and send a data replication packet containing at least some of the information relating to the original copy of the data (Gehani, col. 7, lines 44-46); a slave server to store a copy of the data from the master server, the slave server comprising: a slave service layer to receive the data replication packet from the master service layer and process the data replication packet, and to send information relating to the data replication packet; and a slave user layer to receive the information relating to the data replication packet from the slave service layer, and to store the information in the data replication packet (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46), wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while avoiding extensive delays in the operation of an application program that initiated the commit procedure.

The combination of Gehani and Britton does not explicitly teach the master server sending the slave server an abort message.

However, Fiske teaches a system where the master service layer is adapted to send an abort message to a slave service layer (Fiske, col. 5, lines 37-42).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combination of Gehani and Britton in view of Fiske in order to enable the master server sending the slave server an abort message. One would be motivated to do so in order to allow for the master server to correct potential mistakes encountered during replication.

59. With respect to claim 23, Gehani teaches the invention described in claim 1, including a computer-based system for replicating data over a network, comprising: a master server containing an original copy of the data, the master server comprising: a master user layer to start a data replication process by calling a start method, the master user layer further adapted to send information relating to the original copy of the data; a master service layer, containing the start method to receive the call from the master user layer and the information

relating to the original copy of the data, the master service layer and to create and send a data replication packet containing at least some of the information relating to the original copy of the data (Gehani, col. 7, lines 44-46); a slave server to store a copy of the data from the master server, the slave server comprising: a slave service layer to receive the data replication packet from the master service layer and process the data replication packet, and to send information relating to the data replication packet; and a slave user layer to receive the information relating to the data replication packet from the slave service layer, and to store the information in the data replication packet (Gehani, col. 7, lines 46-60).

Gehani does not explicitly teach a one- or two-phase update.

However, Britton teaches where based on the type of delta, a one-phase update is done with the slave service layer requesting a delta from the master service layer asynchronously (Britton, col. 31, lines 30-65); or a two-phase update is done with the slave service layer receiving a delta from the master service layer, sending a prepared signal to the master service layer (Britton, col. 32, lines 30-57) and then committing the delta if a commit signal is received from the master service layer (Britton, col. 33, lines 27-46), wherein the one-phase update is done to multiple slave servers (Britton, Fig. 1, reference numerals 14, 16, 18; col. 31, lines 30-65).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Gehani in view of Britton in order to enable the use of a one- or two-phase update. One would be motivated to do so in order to provide a process for resynchronizing a commit procedure for protected resources and conversations while

avoiding extensive delays in the operation of an application program that initiated the commit procedure.

The combination of Gehani and Britton does not explicitly teach the master server sending the slave server an abort message.

However, Fiske teaches a system where the master service layer is adapted to send an abort message to a slave service layer (Fiske, col. 5, lines 37-42).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combination of Gehani and Britton in view of Fiske in order to enable the master server sending the slave server an abort message. One would be motivated to do so in order to allow for the master server to correct potential mistakes encountered during replication.

Response to Arguments

60. Applicant's arguments filed 18 May 2007 have been fully considered, but they are not persuasive for the reasons set forth below.

61. ***Applicant Argues:*** Independent claims 1, 36, 40, 41 and 46-51, as amended, include the limitation "wherein the one-phase update is done to multiple servers." This is not shown, suggested or given a motivation for in the cited prior art.

In Response: The examiner respectfully submits that Britton teaches wherein the one-phase update is done to multiple servers (Britton, Fig. 2, reference numerals 52A and 52D; col. 31, lines 30-65). Britton shows the use of multiple servers (different application environments in different systems, 52A and 52D – see Britton, col. 9, lines 18-20). When the application (server) opens the file with the intent of read, the image is considered to be a read only resource. When the application is done reading the file, it closes the file and attempts a one-phase commit. When the shared file resource manager performs the commit as a read-only resource, [it] could discard the image maintained for the application's use. Now, if the application opens the file again, it will see an image of the file which contains all committed updates made by other applications (see Britton, col. 9, lines 56-65). Thus, the other servers may also view the file as read-only and request a one-phase commit when finished. This renders the rejection proper, and thus the rejection stands.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Alicia Baturay whose telephone number is (571) 272-3981. The examiner can normally be reached at 7:30am - 5pm, Monday - Thursday, and every other Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Saleh Najjar can be reached on (571) 272-4006. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Alicia Baturay
July 19, 2007

SALEH NAJJAR
SUPERVISORY PATENT EXAMINER